UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/717,587	11/21/2000	Christopher G. Kaler	777.338US1	8807

41505 7590 01/09/2006

WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION)
ONE LIBERTY PLACE - 46TH FLOOR
PHILADELPHIA, PA 19103

EXAMINER	
VU, TUAN A	
ART UNIT	**PAPER NUMBER**
2193	

DATE MAILED: 01/09/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/717,587	KALER ET AL.	
	Examiner	Art Unit	
	Tuan A. Vu	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 December 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10,14,15,22-31 and 34* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10,14,15,22-31 and 34* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 12/7/2005.

As indicated in Applicant's response, no claims have been amended. Claims 1-10,14-15, 22-31 and 34 are pending in the office action.

Double Patenting

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 1-10, 14-15, 22-31, and 34 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-7, 9-24 of U.S. Patent No. 09/717645 (hereinafter '645) in view of Leblang et al., USPN: 5,649,200 (i.e. Leblang). Although the conflicting claims are not identical, they are not patentably distinct from each other because the instant claims represent obvious variations of the invention recited in the claims of the '645 application. The following are but a few examples of such conflicts.

As per instant claims 1 and 22, '645 claims 2 and 10 also recite setting a start time with a value representing the current time in the data structure; but these '645 claims do not disclose creating a link content data structure; setting a link set reference field

referring to a link set data structure corresponding to set of associated project management objects; and setting an object reference field to refer to the project management object as recited in the instant claims. In an analogous method to control objects versioning, Leblang discloses the setting and creating of link set and link data structure or object reference field as claimed (col. 9, lines 8-56; *configuration record* – Fig. 23; *derived object 500 – Fig. 21*; col. 32, line 55 to col. 39; Fig. 20; *VOB* -Fig. 22; link *530* - Fig. 23). It would have been obvious for one of ordinary skill in the art at the time the invention was made to add the creation of content data structure and reference link set as claimed to the '645 invention because with such link set and data structure, the method of updating database by '645 would be more enhanced and fault-free when separate data structures are created and set to support users update instances and thereby keep the database from being overwhelmed by simultaneous update operations with potential contention issues.

As per instant claims 3 and 24, '645 claims 6, 14 also recite such data structure being a row in a database.

As per instant claims 6 and 27, '645 claims 2 and 10 also recite setting an end time field but do not recite some limitations for which the teachings by Leblang as set forth from above would have rendered obvious, such limitations being receiving a link set identifier, a reference to the managed object, and locating a link content data structure.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

> The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a " useful, concrete, and tangible result" be accomplished. An "abstract idea" when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the "useful arts" when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a "useful, concrete and tangible result".

5. Claim 14 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claim 14 recites a data structure having some non-functional descriptive elements such as first field, second field, third field, fourth field, wherein the 2nd and 3rd field define some range. As such, the claim does not provide any action or interaction between the recited structural elements in order to enable a reasonable interpretation that a concrete, tangible, and useful result is present or yielded based on such interaction or actions taken. The claim amounts to an abstract idea for failing the practical test requirements; and is rejected as non-statutory subject matter.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1, 3-6, 8-10, 14, 22, 24-27, 29-31 and 34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Leblang et al., USPN: 5,649,200 (hereinafter Leblang), in view of Eisenberg et al., USPN: 5,890,166 (hereinafter Eisenberg).

As per claim 1, Leblang discloses a computerized method for adding (e.g. Fig. 19-23; col. 27, lines 50-67; col. 29, line 3 to col. 34, line 8) an association of a project management object (hereinafter PMO) to a set of associated project management objects, where a link set data structure corresponds to the set of associated project management objects, the method comprising:

creating a link content data structure (e.g. *view ...build ... Release* – col. 9, lines 8-56; *configuration record* – Fig. 23), comprising a link set reference field, an object reference field (ORF) (e.g. entry *532*, link 530 - Fig. 20 – Note: entry in CR and link 530 read on link set reference field, and an object reference field, respectively);

adding an association of the project management object to said set of associated project management objects by setting said link set reference field (LSRF) in said link content data structure to a value that refers to a link set data structure corresponding to the set of associated PMOs (e.g. entry *532*, link *530*; Fig. 20; *VOB* -Fig. 22; link *530* - Fig. 23 – Note: link set reference field, an entry stored link content data structure, i.e. *configuration record,* has a value calling for *link 530* – a link set data structure corresponding to more than on objects, such *derived objects 500* from a VOB database of linked derived objects being equivalent to set of PMOs; a LSRF being just the place to store an ORF, and an ORF being itself a pointer variable as cited in the rejection entry 532 and link 530);

setting said object reference field in said link content data structure to refer to the project management object (e.g. an field calling for link *530* - Fig. 23).

But Leblang does not specify creating a start time field in said link content data structure; nor does Leblang teach setting a start time field in the link content data structure to a value representing the current. However, Leblang teaches associating a time stamp with the referred to objects for update (e.g. Fig. 8-9; col. 11, lines 44-51; col. 16, line 49-55). Besides, the associating of time stamp to version control and informing on life scope of versioned objects was a well-known concept at the time of the invention. Further, in a method to control versioning analogous to Leblang's method, Eisenberg discloses setting a start time field in the link content data structure to a value representing the current time (e.g. col. 15, lines 17-20, 27-29) and setting an end time field in the link content data structure to a value representing a most recent version of the object (e.g. e.g. col. 5, lines 39-41). It would have been obvious for one of ordinary skill in the art at the time the invention was made to provide the techniques of setting the start and end time as suggested by Eisenberg to Leblang's time stamp attribute because the recording of time variance or time elapsed related to database update operations enable better understanding of real-world interdependencies of versioned objects, their most current versions, and the dynamic state of their being updated by more than on operators over time (see Eisenberg: col. 14, line 1 to col. 17, line 67).

As per claim 3, Leblang teaches that the link content data structure is a row (e.g. *record 514* – Fig. 20; Fig. 21 – Note: one entry for the record in the derived object table is equivalent to a row).

As per claim 4, Leblang does not explicitly specify that accessing the derived objects for version update, i.e. link set data structure, means referring to a row in a database; but teaches a table or record of the object to modify (e.g. *VERSION OBJECT, additional fields* – Fig. 9), hence implicitly teaches row to such record.

As per claim 5, Leblang disclose include/header files or object code, meta-data and program source, release notes and scripts (e.g. col. 28, line 33 to col. 29, line 31; Fig. 17; record 532 – Fig. 23; *rlsnotes 208* – Fig. 6); hence discloses PMOs as claimed.

As per claim 6, Leblang discloses a computerized method for removing (e.g. *merge, check-in* – Fig. 13-15) an association of a project management object (hereinafter PMO) from a set of associated project management objects, the method comprising:

receiving an identifier for a link set corresponding to the set of associated PMOs (e.g. link *530, derived object 500* - Fig. 22, 23);

receiving a reference to the PMO (e.g. col. 32, line 55 to col. 39; link *530* - Fig. 23);

locating a link content data structure containing the reference to the PMO (e.g. record *514* – Fig. 20; *config rec 530*, link *530* – Fig. 21).

But Leblang fails to specify setting an end time field in the link content data structure to a value representing the current time. But this limitation has been addressed in claim 1 above using Eisenberg's teachings, hence is rejected herein likewise.

As per claims 8-9, refer to rejection of claims 4 and 3, respectively.

As per claim 10, see claim 5.

As per claim 14, Leblang discloses a data structure comprising:

a first field comprising a reference to a link set data structure corresponding to a set of associated PMOs (e.g. entries *523*, link *530* - Fig. 23);

But Leblang does not specify a second field comprising a start time, a third field comprising a end time, wherein the second and third field define a range of time that the target PMO is associated with the set of associated PMOs. But the limitation of having 2 such fields to use as range for determining time of the target PMO has been addressed in claim 11 above.

As per claim 22, Leblang discloses a computer medium having instructions for performing a method for adding (e.g. Fig. 19-23; col. 27, lines 50-67) an association of a PMO to a set of associated project management objects, where a link set data structure corresponds to the set of associated PMOs, the method comprising:

creating a link content data structure (e.g. *view ...build ... Release* – col. 9, lines 8-56; *configuration record* – Fig. 23);

setting said link set reference field in said link content data structure to a value that refers to a link set data structure corresponding to the set of associated PMOs (e.g. entry *532*, link *530*; Fig. 20; *VOB* -Fig. 22; link *530* - Fig. 23 – Note: link set reference field, an entry stored link content data structure, i.e. *configuration record,* has a value calling for *link 530* – a link set data structure corresponding to more than on objects, such *derived objects 500* from a VOB database of linked derived objects being equivalent to set of PMOs);

setting said object reference field in said link content data structure to refer to the project management object (e.g. an field calling for link *530* - Fig. 23).

But Leblang does not specify setting a start time field in the link content data structure to a value representing the current. However, this limitation has been addressed in claim 1 above; and is rejected herein with the corresponding rejection set forth therein.

As per claims 24-26, these claims correspond to claims 3-5; and are rejected likewise, respectively.

As per claim 27, this is a computer medium claim of corresponding claim 6; hence is rejected with the same rationale used therein.

As per claims 29-31, refer to claims 3-5, respectively, for corresponding rejections.

As per claim 32, this is a computer medium claim of corresponding claim 11; hence is rejected with the same rationale used therein.

As per claim 34, see claim 5.

8. Claims 2, 7, 15, 23, and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Leblang et al., USPN: 5,649,200, and Eisenberg et al., USPN: 5,890,166, as applied to claims 1, 6, 14, 22, 27 from above, in view of Reed et al., USPN: 5,862,325 (hereinafter Reed).

As per claim 2, Leblang only teaches link being hard pathname but does not specify object reference field being an URL. However, Leblang teaches hyperlinks to reach versioned data or database (e.g. col. 16, line 57 to col. 17, line 29). Reed, in a communication scheme to enable update of database versioned data analogous to Leblang's version control method using system hard links, discloses use of URL to reach out for version instances of objects (e.g. *http://company.com/commobject3481.cos* -- col. 91, line 27 to col. 92, line 9; Fig. 2). It would have been obvious for one of ordinary skill

in the art at the time the invention was made to provide such URL to locate version database as suggested by Reed, and apply this to Leblang linking method because database accessible across the internet would alleviate extraneous storage resources of the local system or environment under which Leblang's method operates for version update.

As per claim 7, this claim corresponds to claim 2 above, hence is rejected herein using the same rationale as set forth therein.

As per claim 15, see claim 2.

As per claims 23 and 28, see claim 2.

Response to Arguments

9. Applicant's arguments filed 12/7/2005 have been fully considered but they are not persuasive. Following are the Examiner's remarks therefor.

Rejections pursuant to 35 USC §101:

(A) The arguments against the USC 101 rejection are mere repeat of those which had been submitted in the Applicants response filed 3/14/05; in regard to which hereafter is the Examiner's response.

Applicants have submitted that the data structure of claim 14 is a permissible form of patentable subject matter if the claim represents functional descriptive material that becomes 'structurally and functionally related to the medium ... realized through use of technology' (Appl. Rmrks, pg. 2, bottom 2 para). First, in order for functional material to be subject matter for patentability, it must be claimed. Applicants only alleged that the structural elements recited as first field, second field, third field and fourth field in the data structure are functional descriptive material and that they are functionally related to the medium while absent is any hint of action being taken or any execution performed

(upon the recited *fields*) by any form of tangible support in the related field of technology supporting such data structure thus stored. Claiming that a data structure is being persisted in a storage and that such structure has fields clearly amounts to just listing descriptive elements; and from there alleging that all those fields achieve functional matter in light of the very medium used to persist the data structure is nothing but a leap from a pure element listing to performing an action, allegation founded on lack of evidence to support a subject matter that Applicants believe comes from the claim. Reciting descriptive material with total absence of any explicit action take to process such material in a functional interrelationship manner would not be pursuant to what is expected according to the *State Street* bank case wherein a action must be provided so to use/operate on such descriptive material to enable the yielding of useful, concrete and tangible result in the art of interest. Claim 14 as such can be analogized to declaring a system storing a file; and such file has entry 1, entry 2, and entry 3; each such entry storing some data related to time for example. The question is if one user were to put this file into, say, a computer whereas there absolutely no procedure showing how this file is to be utilized, can this file be a product that a user or a skill in the art can buy or make use of as a valuable utility or useful product? And can the invisible procedure be inferred based on an assumption that the file itself signifies automatic functional usefulness just by the fact that it can be stored and fetched by a computer file system? Whereas 35 USC §101 requires that the subject matter has to be substantial, specific and credible, the argument from Applicants asking one skill in the art to accept existence of a fact that is not specifically explicit or substantial amounts to stretching or reaching out for something

that is too weak to be given credible existence to. Therefore, the rejection will stand as set forth.

Rejections pursuant to 35 USC §103(a):

(B) The arguments against the USC 103(a) rejection are mere repeat of those which had been submitted in the Applicants previous response in which the issue that the Examiner incorrectly analogized *link set data structure* and *object reference field* with Leblang's link 530 had been raised; in regard to which (Appl. Rmrks, pg. 8, middle, pg. 9, top) the Examiner's response would be again as follows.

Applicants have submitted that Leblang link 530 is being analogized to 2 different elements of the claimed invention, i.e. object reference field and link set reference field; hence Leblang teaching is flawed. The rejection has explained that an entry in the CR of Fig. 20 reads on link set reference field (LSRF); and that link 530 reads on object reference field; and that a data structure for corresponding VOB derived objects reads on link set data structure, i.e. link 530. Now the problem is to prove whether the element recited as 'link set data structure' (LSDS) distinguishes significantly from what is recited as 'object reference field' (ORF). The claim teaches (i) that a link set data structure (LSDS) is for pointing/corresponding or referring to the set of associated project management objects (PMO); and this reference is made via the setting of an entry in the link content data structure (LCDS) to refer to this LSDS. The claim further recites that the LCDS also contains a ORF, and that (ii) a ORF in the link content data structure (LCDS) is set to refer to a LSDS corresponding to the above set of managed objects. Since from (i) the LSDS is for enabling an LSRF entry of the LCDS to correspond/relate to some PMO set; it is understood that each time that there is reference or correspondence

or inter-association between a LSDS and a field in LCDS, from (ii) a field entry in this LCDS is set to refer to (i.e. a pointing data structure being instantiated) a LSDS which is associating to the set of PMO. Hence, by virtue of relating (i) to (ii), it is understood a ORF in said LCDS is also set to correspond/point to some PMO. If an ORF is also used to correspond with or refer to a PMO in a set of PMO; it is thus plausible to interpret that by having a LSDS corresponding or pointing to a set of PMO via entry setting of a LCDS, the LSDS amounts to just having data structure being set for such correspondence or association, i.e. mapping exactly what a ORF is purported to do, i.e. to corresponding or referring to some PMO, making the ORF same in functionality as a LSDS. Therefore, there is no significant difference between what a link set data structure does (corresponding to a set of PMO) and what a object reference field does (set to refer to a PMO), considering the fact that corresponding or pointing to a set of PMOs inherently encompasses the concept of being set to point to any one element of such PMO set. Hence, Leblang's *link 530* for being but a pointing link referring to a PMO is analogous to a LSDS which by its very nomenclature, entails a data structure (data structure being a very broad term) set to link to Leblang's derived objects in the VOB – some PMOs, which is equivalently performed by the ORF. The ORF is also such field or a pointer being instantiated in Fig. 21 to point to a object in a VOB; making link 530 suited to read on both a LSDS and a ORF.

Furthermore, the claim does not make it clear that by setting a value to a LSRF to refer to a LSDS, such setting would provide a distinct reference than that effected via the setting of a ORF to refer to the PMO. As understood from (i) and (ii) above, both settings are equally purported to make linkage of a field in the LCDS to a set of PMO via

an instance of LSDS. The rejection has pointed to an ORF being an instantiated pointer in a LCDS when a LSRF is but an address for such pointing structure; and when such address is instantiated to store such pointer, a value of LSRF is set. When the pointer in the LCDS is set to point to a object in a VOB, such pointer would be the LSDS --or link 530 in Fig. 23; and by virtue of the instantiated pointer from the setting of LSRF, pointer 530 also maps to an ORF, both ORF and LSDS is set to refer to one set of PMOs; with the ORF here referring to link 530 which is the LSDS.

(B) Applicants have disagreed (Appl. Rmrks, pg. 9, 3rd para) on the Examiner's introduction of a 'correspondence' (from: a field entry in this LCDS is set to refer to -- a pointing data structure being instantiated- a LSDS corresponding to the set of PMO) because such 'correspondence' is not claimed. What is claimed is setting a field in a LCDS to refer to a LSDS which is already corresponding to a set of PMOs. According to a broad interpretation of the above, there is relationship between an existing LSDS and a field in a LCDS – or LSRF - being set to refer to this LSDS; and this is perceived as an referencing association or pointing type of correspondence. The interpretation leads to the rationale explaining why having a ORF and a structure like a LSDS would amount to doing the same functionality; as set forth above. Arguments to the effect that no change needs to happen in the LSDS would become moot because the rejection does not address a limitation in which a LSDS needs to incur a change within.

(C) Applicants have submitted that examiner's contention that a field entry in the LSDS must be changed to point to a PMO set (Appl. Rmrks, pg. 10, 2nd para). This argument falls under Applicants' perception that Examiner introduces a limitation as to make change to a LSDS each time there is a reference being set in the LCDS; which is

not true according to section A above. Leblang's link 530 refers to a set of PMO objects; and this fulfills the claimed limitation as far as 'a link set data structure corresponds to the set of associated project management objects' is described in the claim because such correspondence (i.e. *corresponds to the set* ...; *corresponding to the set of* ...) is not recited in more precise terms as to enforce on a persistency (or unchanging nature) of the link structure or a time-based durability of such LSDS. Broad interpretation has it that Leblang's *link 530* like that proffered in the rejection would enable correspondence to an object in the set of VOB objects of Fig. 24.

The arguments are not persuasive and for the most part are repeat of previously submitted arguments including those submitted in Applicant's After-Final request for consideration; and the claims stand rejected as set forth in the rejection.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence – please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Tuan A Vu

Patent Examiner,
Art Unit 2193
January 05, 2006